

EJB기반 웹 컴포넌트 모델 분석 및 설계에 관한 연구

*이 돈 양, **송 영 재
경희대학교 전자계산공학과
전화 : 031-201-2946 / 핸드폰 : 011-9017-3709

A study of web component model analysis and design using EJB driven

Don-Yang Lee, Young-Jae Song
Dept. of Computer Engineering, KyungHee University
E-mail : dylee6211@hanmail.net

요 약

최근에 인터넷 및 웹의 급속한 발전과 더불어 S/W 개발의 새로운 방법으로 컴포넌트가 대두되고 있다. 이는 빠른 시간 안에 원하는 소프트웨어 제품을 적은 비용과 최소한의 노력으로 생산할 수 있어 효율적인 시스템 개발 및 소프트웨어 재사용에 많은 이점을 가질 수 있다. 그러나 현재 사용되고 있는 COM, CORBA 등은 프로그램 작성뿐만 아니라, 분산 시스템 하의 서로 다른 플랫폼에서 운영상의 문제점들이 발생하게 되었다.

본 논문에서 제시한 EJB 방식의 아키텍처 시스템은 서로 다른 네트워크 환경과 이기종의 시스템에 존재하는 프로그램 객체들을 동일 시스템에서 사용하는 것처럼 통합을 가능하게 해주고 있다. 그리고 사례연구를 통하여 웹 컴포넌트의 모델분석과 설계를 하여 빈 (Bean)을 생성하였으며, 결론으로 COM+, CORBA Component(CCM), EJB기반 컴포넌트를 비교하였다.

현재는 COM나 CORBA 등이 사용되고 있으나 COM은 명세가 애매하고 이해하기가 어렵고, CORBA는 오브젝트 배분의 비표준화와 제한적인 오브젝트 함수로 인하여 문제점이 있다. 또한, 분산 시스템의 운영방식에서는 이기종의 컴퓨터들이 이용되면서 서로 다른 플랫폼을 사용하기 때문에 운영상의 문제점들도 발생되었다. 그러나 새로운 대안으로써 개발된 EJB 방식의 아키텍처 시스템은 CORBA를 기초로 하여 작성되었으며, 객체지향언어 구조로 JAVA로 구현되면서 분산환경에 적합하고, 이 기종 시스템에 존재하는 프로그램 객체들을 동일 시스템에서 사용하는 것처럼 통합을 가능하게 해주고 있다. 본 논문에서는 EJB 기반으로 생성된 컴포넌트를 웹상에서 운영되도록 하기위해서 시스템을 분석하고 설계하였다. 사례연구를 위해 웹에서 운영되는 강의 사이트를 구축하였으며, 컴포넌트의 모델분석 및 설계에 UML을 이용하였다.

결론으로는 COM+, CORBA Component(CCM), EJB 기반 컴포넌트를 비교하면서 앞으로의 연구방향도 제시하였다.

I. 서론

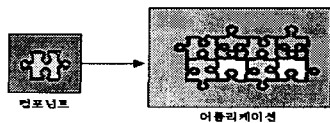
현대의 소프트웨어 개발에서는 빠른 시간 안에 원하는 제품을 적은 비용과 최소한의 노력으로 생산해 내는 것에 많은 비중을 두고 있다. 이를 충족시키기 위해서 컴포넌트기반 소프트웨어 개발방법론이 대두되었고,

II. 관련 연구

2.1 컴포넌트

컴포넌트는 재사용이 가능한 독립적인 기능을 가진 어플리케이션의 모음으로[1], <그림 1>에서와 같이 컴포넌트는 대형의 어플리케이션 프로그램을 개발하는데 사용되어지는 조립형태의 부품으로 생각하면 된다.

컴포넌트 패러다임의 중요한 관점으로 그 근원을 살펴볼 필요가 있다. 첫째로 Catalysis는 컴포넌트 기반으로 발달된 UML에 관련되어 표기되어온 최초의 방법중의 하나이다.[2] 이는 "modeling concepts", "levels of modeling", "principles"등 기본적으로 세 가지로 구분되는 구성을 가지고 있다. 두 번째로 Business Process Modeling 강조하는 방법인 Select Perspective[3]있고, 세 번째로 UML 컴포넌트가 있다. 이 방법은 Cheesman과 Daniels 가 제안한 것으로 Catalysis, RUP, Syntropy의 강한 영향을 받았다.[4]

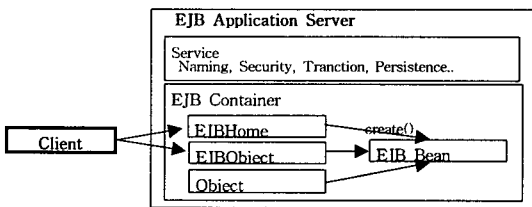


<그림 1> 컴포넌트 그림

현재는 수많은 컴포넌트 모델이 사용되고 있으나 주로 사용되고 있는 것으로는 OMG's의 CORBA Component Model(CCM), Microsoft's (D)COM/COM+ family, SUN Microsystems의 JavaBens 와 Enterprise Java Beans가 있다.[5]

2.2 EJB(Enterprise Java Beans)

분산처리 시스템의 등장으로 인하여 이 기종 시스템 간의 프로그램 분산이 이루어져 시스템 및 프로그램의 운영에 많은 향상을 가져왔다. 그러나 이 기종 시스템 간의 통합의 어려움이 발생하여, 이를 해결하기 위한 방법으로 CORBA, (D)COM/COM+, EJB등이 미들웨어로 등장하였다. 이는 서로 다른 네트워크와 시스템 간의 환경에서 객체들을 이용하는데 동일한 시스템처럼 해주고 있다. 그리고 이중 EJB는 분산환경에서 가장 효과적인 시스템운영을 지원하고 있다. <그림 2>와 같이 개념적으로 EJB는 서버 측 Java 컴포넌트에 대한 프레임워크이며[6], 이는 EJB 컴포넌트, EJB 컨테이너, EJB 어플리케이션 서버로 구성되어있다.[7]

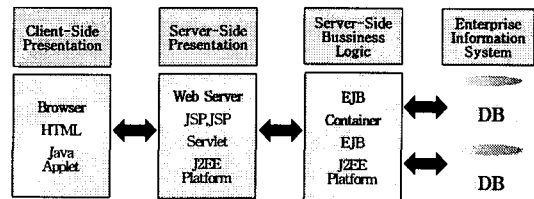


<그림 2> EJB 아키텍처

2.3 J2EE 아키텍처

J2EE(Java2 Platform Enterprise Edition)는 다층의

기업용 어플리케이션 발달에서 표준을 정의하고 있다. J2EE는 컴포넌트의 표준화, 모듈화 기초와 이를 이용한 완성된 컴포넌트를 보급하고, 복잡한 프로그램의 작성 없이 자동적으로 작동되는 많은 어플리케이션의 개체들을 사용함으로써 기업용 어플리케이션 개발을 단순화 하였다. 이는 단순화, 간편화, 스�কে라빌리티, Legacy 통합의 기능을 가지고 있는 기업용 솔루션개발에 대한 플랫폼이다[8]



<그림 3>J2EE 어플리케이션 모델

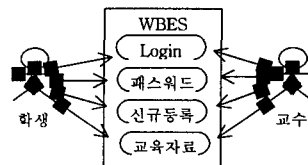
III. 웹 컴포넌트 모델 분석 및 설계

3.1 UML 기반 컴포넌트 모델링 작성

본 논문에서 사례연구로 설계된 프로그램의 이름을 WBES(Web Based Education System)로 하였으며 컴포넌트 생성에 관련된 모든 것들을 WBES에 적용시켰다. 그리고 컴포넌트기반 시스템설계는 UML의 표준화 표기법을 이용하였다.

3.1.1 유즈케이스 다이어그램(Use Case Diagram)

유즈케이스는 시스템구축에서 전체적인 요구사항들을 분석하는데 이용되고 있으며, 여기서는 <그림 4>와 같이 교수와 학생에 대한 2개의 Actor를 사용하였다.



<그림 4> WBES 유즈케이스 다이어그램

3.1.2 유즈케이스(Use Case) 명세서

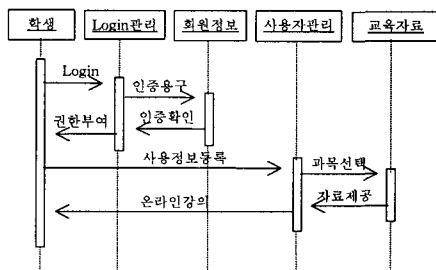
WBES를 설계하기 위해서 2개의 액터(Actor)를 설정하였다. 유즈케이스 명세서에서는 학생과 교수가 시스템과 어떻게 작용하는 지를 알 수 있게 해주며, 이를 바탕으로 <표 1> 같이 시나리오(Scenario)작성과 시스템에서 추가적인 확장(Extension)과 변이(Variation)를 기술할 수 있도록 하고 있다.

<표 1> 학생에 대한 유즈케이스 명세서

이름	: WBES Login Check
initiator	: 학생
목적	: WBES 사용가능 여부
시나리오	: 1. Actor는 WBES를 사용하기위해 Login 한다. 2. WBES는 학생의 사용가능여부를 확인한다.
확장	: 2. WBES는 학생의 ID와 Password를 Check한다. a. 신규 회원등록을 한다. b. 종료

3.1.3 시퀀스(Sequence) 다이어그램

<표 1>의 학생에 대한 시나리오를 바탕으로 <그림 5>의 시퀀스 다이어그램을 작성하였다.



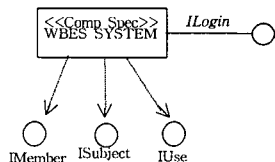
<그림 5> WBES 시퀀스 다이어그램

3.2 컴포넌트 모델링

시스템 구축에서 요구되는 각 부분의 분석을 통하여, 현재 개발범위와 추후 추가되는 부분까지 단계적으로 분석하고 설계하여 최적의 방법으로 시스템의 전체적인 구성과 개발의 용이성, 시간의 절약, 예산의 절감 등을 실현할 수 있도록 컴포넌트 모델링을 하였다.

3.2.1 시스템 컴포넌트 명세

<그림 6>에서 IMember, ISubject, IUse들 간의 의존성 관계를 보여주고 있듯이 시스템 컴포넌트 명세는 하나의 컴포넌트에 연결된 인터페이스들의 구성을 명세하고 있다. 이들 사이의 정확한 의존관계는 컴포넌트의 상호작용 분석 시 검증될 수 있다. 그리고 이와 같이 초기단계의 컴포넌트 명세를 정의함으로써 컴포넌트 아키텍처를 형성할 수 있는 장점이 있다.

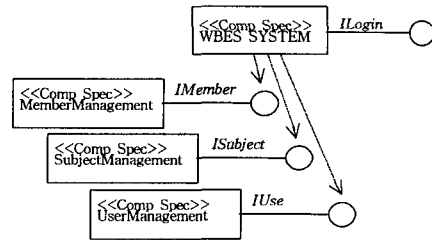


<그림 6> System Component 명세

3.2.2 컴포넌트 명세 아키텍처

컴포넌트는 시스템에서 배포의 단위가 되며 또한 교

체의 단위로 사용된다. 그리고 수정이나 교체에서도 컴포넌트 단위로 이루어지므로 일반적으로 컴포넌트에 대한 명세를 정의함으로써 초기의 컴포넌트 아키텍처를 구성할 수 있고, 이를 통하여 컴포넌트의 상호작용을 분석할 수 있는 구조적인 기초를 다질 수가 있다.



<그림 7> Component 명세 아키텍처

IV. EJB 웹 어플리케이션 시스템 구현

4.1 빈(Beans) 생성

EJB에는 세션 빈(Session Bean)과 엔티티 빈(Entity Bean)인 두 가지의 빈(Beans)이 있다. 빈은 재사용이 가능한 컴포넌트이며 세션빈과 엔티티빈의 차이점은 세션빈은 공유의 데이터를 액세스하고, 클라이언트에 공유 불가능하며, 클라이언트확장, 퍼시스턴스를 갖지 않은 특징이 있으나 엔티티빈은 특정개념의 공유데이터를 표현할 수 있고, 클라이언트에 공유가능하며, 데이터베이스 확장, 퍼시스턴스를 가질 수 있어 두개의 차이점을 비교할 수 있다.

본 사례연구에서는 신규학생 사용자등록에 대한 EJB컴포넌트 작성을 위하여 엔티티빈의 CMP를 이용한 퍼시스턴스를 EJB 컨테이너가 자동으로 관리하도록 하였다.

4.2 컴포넌트 생성

컴포넌트를 작성하기 위한 요구분석 작업을 마치면 <표 2>과 같이 각각의 프로그램을 작성하고, 엔티티빈을 생성하기 위해서 데이터베이스 테이블을 셋팅시키다. 그리고 엔터프라이즈 빈과 JNDI를 이용한 웹 컴포넌트를 생성한 후 클라이언트에서 생성된 컴포넌트를 이용한다. <그림 9>에서는 EduBeanJAR 빈과 EduClientWAR 웹 컴포넌트의 생성을 보여주고 있다.

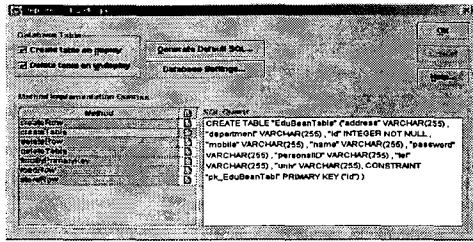
<표 2> 빈 생성 프로그램목록

프로그램종류	프로그램이름
리모트 인터페이스	EduRmt.java
홈 인터페이스	EduHome.java
CMP 엔티티 빈클래스	EduBean.java
프라이머리키 클래스	EduPK.java
클라이언트 입력	EduClient.html
클라이언트 출력	EduClient.jsp

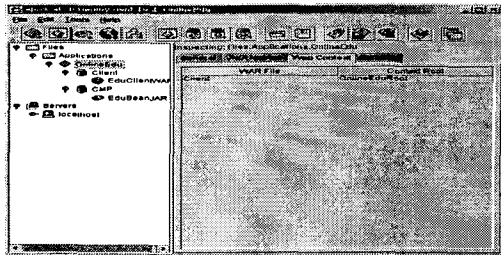
V. 결론 및 향후 연구방향

본 논문에서는 컴포넌트 기반의 분산객체 시스템을 구현하기 위해서 EJB 아키텍처를 사용하였으며, 내부적으로 객체간의 통신 수단으로 자바의 RMI(Remote Method Invocation)기술을 이용하면서 프로그래머가 시스템을 개발하는 데 큰 어려움이 없이 신속하게 컴포넌트를 생성하도록 하였다.

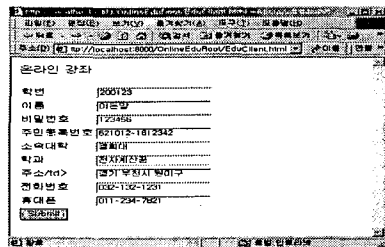
EJB가 상업적 목적으로 이용되는 대규모의 시스템을 구축하는데 매우 유용하게 사용되기 때문에 규모가 작은 어플리케이션 시스템을 구축하는 데에는 빈(Bean)을 설계하고 구성하여, 생성하는 데 시간이 많이 걸리고 또한 프로그램작성의 어려움 때문에 그 실효가 떨어질 수 있지만, 사례연구를 통해서 작은 단위의 빈(Bean)과 컴포넌트를 생성하면서, 이를 기초로 추후 대규모의 어플리케이션 시스템을 구축하는데 토대를 마련하고자 하였다. 그러나 단점으로는 EJB 기반 컴포넌트 시스템을 구현하기 위해서는 자바 관련 언어의 습득이 필수적이고, 컴포넌트를 사용한 어플리케이션 운영면에서는 네트워크오버헤드, 유지보수등 몇 가지 문제점들이 발생되어 추후 많은 연구가 이루어져야 할 것이다.



<그림 8> 데이터베이스 테이블 생성



<그림 9> 빈 과 웹 컴포넌트



<그림 10> 클라이언트입력

<표 3> 아키텍처 특성비교

컴포넌트환경	플랫폼의존성	언어의존성	인터페이스상속	추상화
COM+	Windows	없음	없음	낮음
CORBA	없음	Java	다중	높음
EJB	없음	Java	다중	높음

<표 3>에서는 COM+, CORBA, EJB를 각각의 특성별로 비교하였다. 그러나 CORBA는 오브젝트함수의 비표준화와 제한적인 오브젝트함수로 인하여 문제점이 있어 일반적으로 EJB기반에서 컴포넌트를 생성하는 것이 낫다.

6. 참고 문헌

- [1] George T. Heineman and William T. Council "Component-Based Software Engineering", 2001.
- [2] D'Souza, D. F. and Wills A. C., Objects, Components and Frameworks with UML: The Catalysis Approach, Addison-Wesley, 1998.
- [3] Cheesman, J. and Daniels, J., UML Components: A simple Process for Specifying Component-Based Software, Addison-Wesley, 2000.
- [4] Allen, P. and Frost, S., Component-Based Development for Enterprise System, Applying the SELECT Perspective, Cambridge University Press/SIGS, Cambridge, 1998
- [5] Tim Souder, Spiros Mancoridis, and Maher Salah "Form: A framework for Creating Views of Program Executions", IEEE Software, 2000
- [6] R.Manson-Haefel. Enterprise JavaBeans. O'Reilly & Associates, 2nd edition, 2000.
- [7] 김수동, Enterprise JavaBeans(EJB) 기반의 컴포넌트 프로그래밍. 정보처리 제7권 제4
- [8] The Java[tm] 2 Platform, Enterprise Edition (J 2 E E [t m]) , [http://www.sun.com/products-n-solutions /software/oe-platforms/java2ee.html](http://www.sun.com/products-n-solutions/software/oe-platforms/java2ee.html)